R-09-022-039

..A..ONAL BUREAU OF STANDARDS REPORT

..S ..OJECT
..30121
.230420

NBS REPORT
8665

December 29, 1964

MAGIC

A MACHINE FOR AUTOMATIC GRAPHICS INTERFACE TO A COMPUTER

by

Don E. Rippy

GPO PRICE $ ______

OTS PRICE(S) $ ______

Hard copy (HC) 3.00

Microfiche (MF) .50

N65-25010 (ACCESSION NUMBER)

(THRU)

58 (PAGES)

1 (CODE)

CR 62897 (NASA CR OR TMX OR AD NUMBER)

08 (CATEGORY)

FACILITY FORM 602

Information Technology Division

NBS

U. S. DEPARTMENT OF COMMERCE
NATIONAL BUREAU OF STANDARDS

FOREWORD

The Computer Technology Section is currently engaged in an extensive program to develop advanced techniques for improving user communication with large ADP systems. This program is an outgrowth of a number of projects in which the Section has been called upon to assist other agencies in the solution of a variety of data processing applications. These projects have ranged from data acquisition, using digital data-logging equipment, to the development of an electronic data interchange center constructed around an internally programmed computer of unique organization.

These projects have provided contact with many kinds of data processing problems and have afforded an opportunity for consultation with regard to broader on-going tasks of a more complex nature. These tasks generally involve such functions as command and control, design and mapping, updating and utilization of active files, editing, and information retrieval. The implementation of these functions requires that data processing capabilities be made accessible to users who are essentially task-oriented, rather than machine-oriented. This in turn requires the development of simple, effective techniques for achieving communication between the data processing system and the users. This requirement points toward the need for advanced display and input devices designed to present data in easily assimilated form, and to permit data to be entered and manipulated in a manner familiar to the user.

This report describes a machine which has been developed within the Computer Technology Section as a research tool for the investigation of man-machine communication techniques. This machine has been designated MAGIC (Machine for Automatic Graphics Interface to a Computer). This machine combines large-diameter cathode-ray displays with a specially designed programmable digital computer. It is designed as a remote display station and is intended to be connected to a large ADP system via communication lines. Extensive design effort has been devoted to removing from the

ADP system the time-consuming and repetitive tasks of display regeneration and manipulation, and to minimize the limitations introduced by the communication lines. Particular emphasis has been placed on establishing the proper balance between hardware and software functions.

MAGIC was originated in August 1964 and completed in February 1965. It is currently being used to conduct experiments and perform demonstrations in order to better define the optimum characteristics for equipment of this type. The development and programming of MAGIC has been supported by the National Bureau of Standards and by the National Aeronautics and Space Administration under Contract R-09-022-039.

ACKNOWLEDGMENTS

The over-all development of MAGIC from conception to present operational status involved contributions from every member of the Computer Technology Section. However, the author would like to particularly acknowledge the contributions of the following individuals within the section:

J. A. Cunningham and P. Meissner for their over-all contributions to the concepts and design of MAGIC.

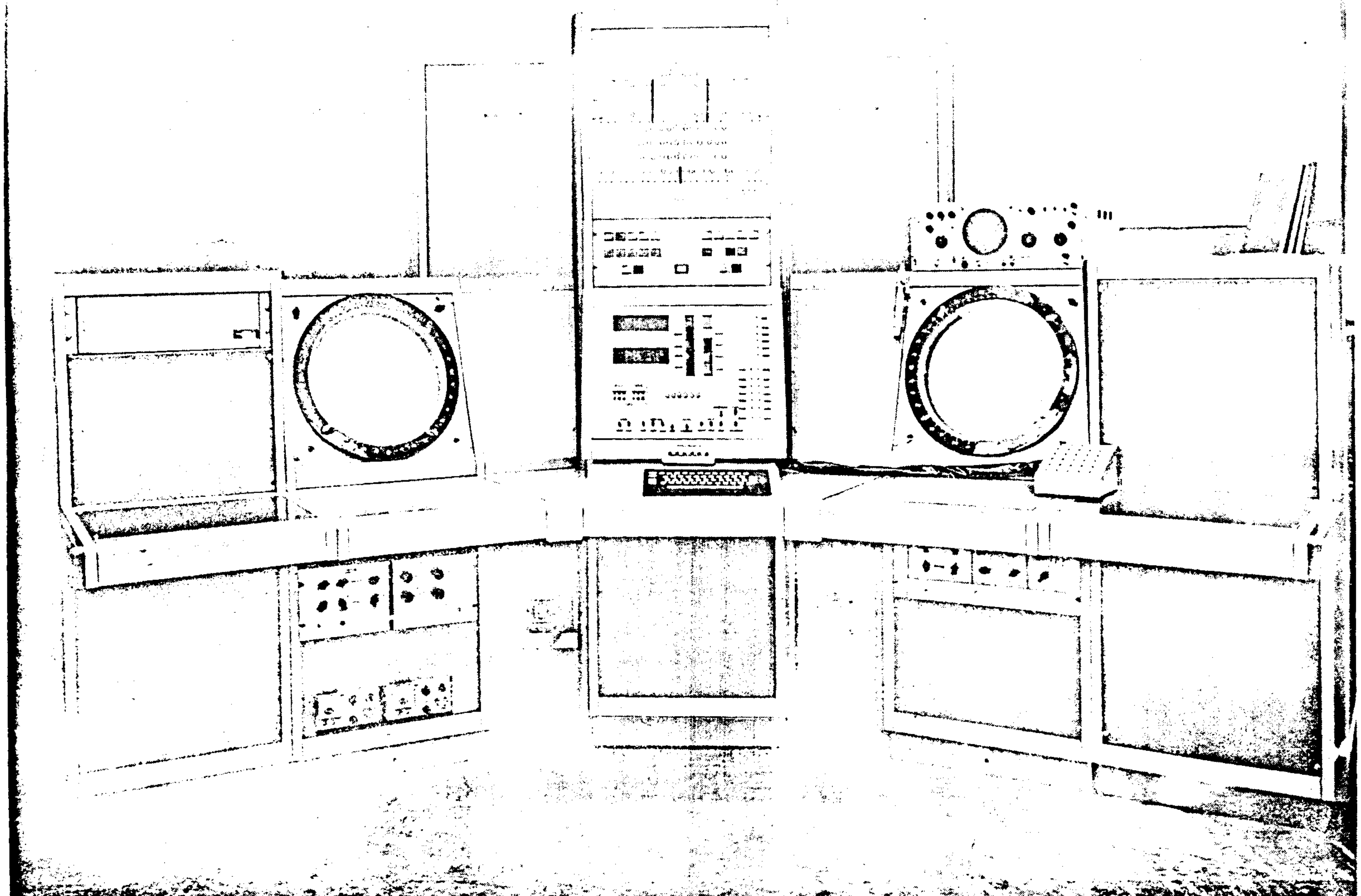
D. E. Humphries for the analog aspects of the hardware design of MAGIC.

L. J. Boezi and O. Hall for their development of the hardware for the locator and the drum memory, respectively.

W. A. Barr, W. H. Basye, R. L. French and V. Ordaz; without whom the construction of MAGIC from available surplus hardware could not have been accomplished.

Editorial and drafting services were most ably provided by P. Meissner and J. Warme.

CONTENTS



MAGIC - A Machine for Automatic Graphics Interface to a Computer

MAGIC

A MACHINE FOR AUTOMATIC GRAPHICS INTERFACE TO A COMPUTER

by

Don E. Rippy

I. INTRODUCTION

This report presents the programming and operating procedures for MAGIC. The purpose of this machine is to serve as a research tool for studies in the area of man-machine communications. It is intended to be used ultimately as a peripheral or remote device as part of a large data processing system. In such a system the high-speed central processor would be available for performing detailed arithmetic computations and other involved manipulations. Thus, the function of MAGIC is to serve as a data entry and display device, permitting the user to operate with graphical information in a familiar visual form.

MAGIC consists of two large cathode-ray display consoles joined together with a specially designed data processor organized around a magnetic-drum memory. A keyboard is included for entering alphanumeric information, and a number of pushbuttons and switches are available which can be assigned a variety of functions. A light pen is available for use with one of the display consoles which permits line drawings and other graphical displays to be prepared. The light pen may also be used to identify portions of a display for manipulation by the data processor.

The data processor has been designed to provide a constant refresh rate for the displays at the rotation rate of the magnetic drum (30 revolutions per second). Displays are generally represented by lists of data, and display manipulations can best be achieved by techniques of list processing. Hence, the data processor is organized as a list processor, utilizing three lists for operating the display, plus a fourth list for program manipulations. These lists are contained in specially-arranged pairs of tracks on the drum which permit block transfers, insertion and deletion of data, and other versatile manipulations. The hardware has been designed to perform many functions that might otherwise require considerable programming effort and time.

A basic block diagram of MAGIC is shown in Fig. 1. The memory consists of a medium-sized magnetic drum divided into two sections:

1) General Memory, consisting of 90 channels of 128 12-bit words each.

2) Display Memory, consisting of 4 "channels" consisting of 128 12-bit words each. These channels are designated W, X, Y and Z. Their functions will be described later.

The general memory includes the input-output channel which is internally addressable in the same manner as any general memory channel and is also addressable from an external source. Such external activation of this I/O channel takes priority over internal activation. At present, the only input or output to MAGIC is via pushbutton control of the I/O register and the I/O channel, or via the keyboard linked to the display channels. The details of these I/O operations are discussed in sections V-D, VI and VII. Also included in general memory are the X', Y' and Z' channels. These act as the memory and data source for the secondary display unit but otherwise function as normal general memory channels (see Fig. 1).

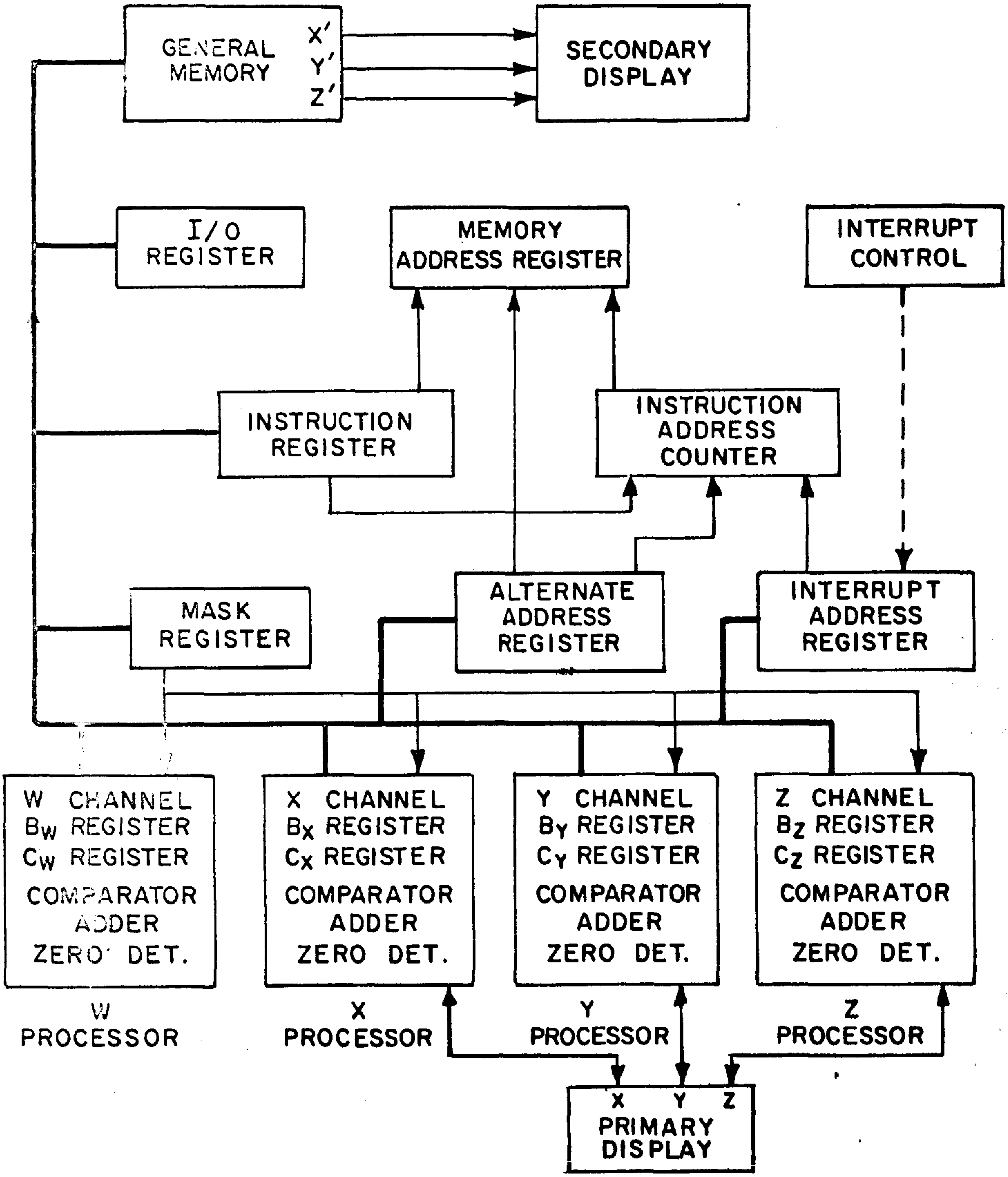


FIGURE 1. MAGIC BLOCK DIAGRAM

MAGIC functions as a single address machine. Since complex mathematical or manipulative operations are not required within MAGIC, the word format is unsigned and without parity. As a result, numerical data is assumed to be positive; the range being 0 to 7777_8 (or 4095_{10}). The basic data word format is shown in Fig. 2.

Bit Position:	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Binary:	0	1	0	1	0	1	1	0	0	0	1	1
Octal:	2			5			4			3		

Figure 2. Basic Data Word Format (single length)

II. MAGIC OPERATION SEQUENCE (control processor)

The operational sequence of MAGIC may be considered to be a repetitive cycle of two "phases". Referring to Fig. 1: During phase one an instruction is fetched from memory and placed in the instruction register. The memory address for the location of this instruction is assumed to be previously loaded into the memory address register manually or at the end of the previous phase two cycle. During phase two the machine performs the operation appearing in the instruction register as a result of a previous phase one cycle. The memory address used during phase two is loaded into the memory address register at the end of the previous phase one cycle. Phase one and phase two are alternately repeated throughout the progress of the program being performed until halted either manually or by programming.

III. REGISTERS

There are eight registers within the control processor portion of MAGIC which are of importance to the operator.

1) Memory Address Register (MAR): All memory addressing (with the exception of the external I/O channel addressing) is performed via this register. Its contents consists of the general memory channel and sector as well as bits designating W, X, Y, Z. The sector address can apply to any general memory channel or W, X, Y or Z, or both, depending on the instruction being performed.

2) Instruction Register (IR): This double length register contains the operation and the memory address of the instruction to be performed. The operation code of the instruction is decoded for execution from the contents of this register.

3) Instruction Address Counter (IAC): The general memory address (channel and sector) of the next instruction appears in this register. The sector portion of the IAC normally advances by two before each instruction is called for since the IR is a double length register. Since instructions may reside only in the general memory, there is no provision for W, X, Y or Z in the IAC.

4) Alternate Address Register (AAR): This register is used for storing an alternate address for entry into the MAR when the alternate address code is detected in the IR (see Section V-A).

5) Mask Register (MR): Contains the masking data for scan operations (see Section V-J).

6) Interrupt Address Register (IAR): Depressing one of the interrupt controls (see Section VI-C) loads a sector address into bit positions 2^0 - 2^6 of this register. At the end of this operation, the sector address is then loaded into the IAC for use during the following phase one cycle (see Sections V-D and VI-B3e).

7) Input-Output Register (IOR): This register is used as the main data communication link between external I/O equipment and the general memory. Its operation is discussed in greater detail in Sections VI-B2 and VII-A2.

8) Memory Word Counter (MWC): Although this register is not available to the operator, its existance is basic to the operation of the machine. Operated by the machine's basic synchronizing circuitry, it contains the time-varying sequence of sector addresses and, in conjunction with the MAR, dictates the periods in time for which a memory sector specified in the MAR is available for a phase one or phase two operation. Other functions of the MWC are discussed in Section VI-B3d.

The registers connected with the display memory which are available to the operator are designated B_w, C_w, B_x, C_x, B_y, C_y, B_z and C_z. These registers are addressable by programming (see Sections V-E and F). Their functions are described in the following section.

IV. DISPLAY PROCESSOR FUNCTIONS

The blocks labeled X, Y and Z in Fig. 1 represent the display processors of MAGIC. X and Y processors furnish the coordinate information to the display. The Z processor furnishes control functions to the display. W is of identical construction but is associated with the control processor.

These display processors are designed to operate on display data in list form and are controlled by the control processor. Consequently they may be considered as small, subordinate list processors. A block diagram of one of the display processors is shown in Fig. 4. Each display processor consists of a continuously active A register (with zero detector) from which data is taken for display, a B register, a C register, a serial binary full adder, and a masked comparator. Data is provided to the primary display directly from the A registers. The remaining circuitry is activated by the various instructions. Coordinate data is received from the locator via the B_x and B_y registers. Z data is received in B_z from the keyboard and the Z control switches on the display control panel (see Section VI-B3).

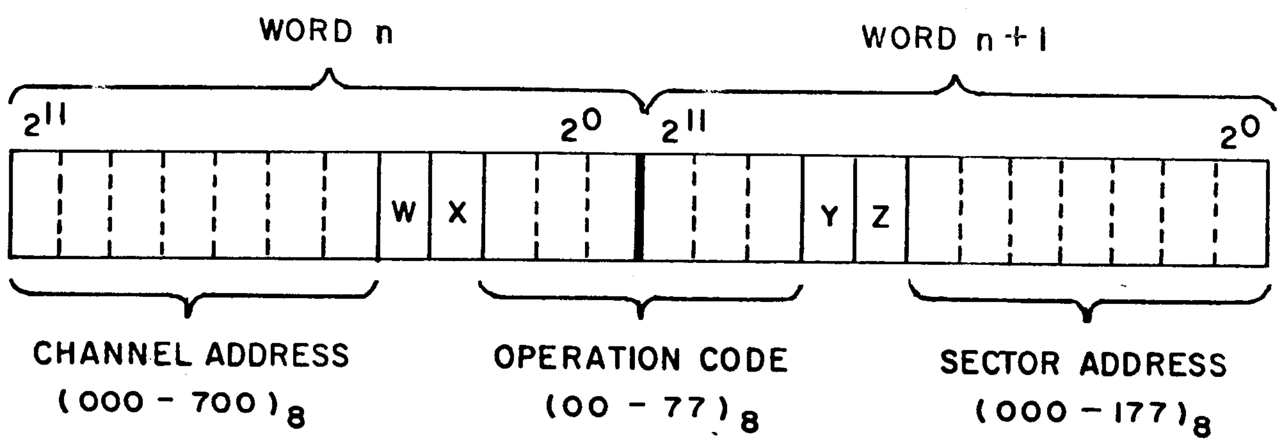


FIGURE 3. INSTRUCTION WORD FORMAT (Double Length)

V. MAGIC INSTRUCTIONS

An instruction is entered into memory as two words (Fig. 3). The first or high order word of the instruction contains the most significant operation code digit. This word usually exists in the even numbered sectors in a given channel. The least significant operation digit is contained in the low order word of the instruction (see Figs. 3 and 3a). Instructions are usually written in octal notation combining bits W, X, Y and Z as part of the octal digit covering their respective bit positions. Thus, to fill the C_x register from sector 72 of channel 114 would be written as:

	word n (even)				word n + 1 (odd)			
Instruction:	1	1	5	2	2	0	7	2
	channel (includes W and X as required)			operation		sector (includes Y and Z as required)		

Figure 3a.

A sample programming sheet appears in appendix B. The individual instructions and their explanations follow. The letter X appearing in any of the numerical examples to follow indicates unused octal digits or binary bits.

A. Alternate Addressing

Sector 177_8 (127_{10}) is used for alternate addressing. If 177_8 appears in the sector portion of the IR as a result of a phase one cycle, the contents of the sector portion of the AAR (AAR bits $2^0 - 2^6$) are transferred to the MAR instead of the sector 177_8 in the IR. The 177_8

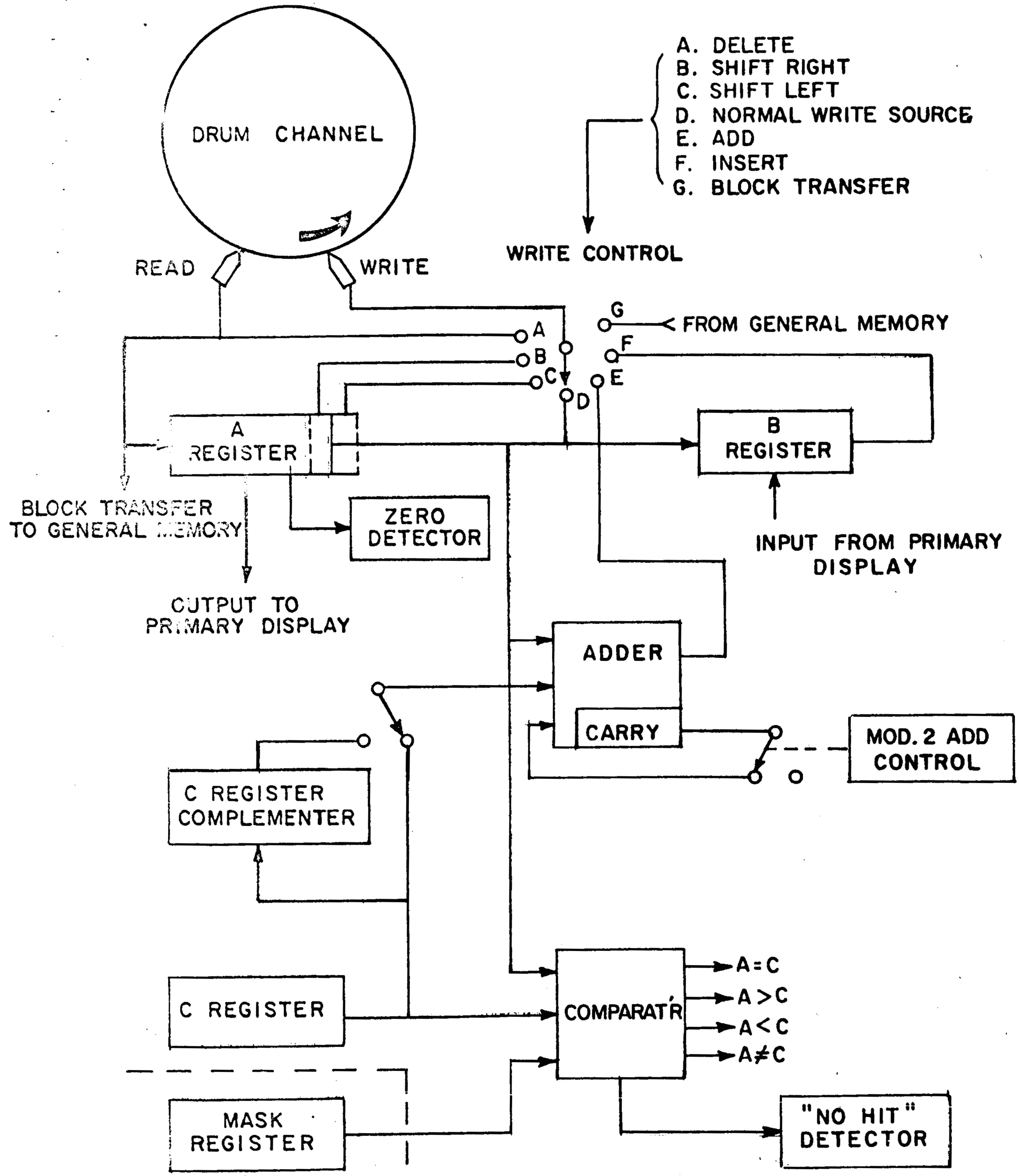


FIGURE 4. W, X, Y, or Z SUBORDINATE LIST PROCESSOR

remains in the sector portion of the IR. The alternate addressing is accomplished at the end of the phase one cycle, and may be applied to all instructions except the CONTROL instructions (see section V-D).

The remainder of this section is devoted to the explanation of the instruction repertoire of MAGIC. An instruction list is provided in appendix A for quick reference.

B. Block Transfers

Transfers of whole channels, parts of channels or single specified sectors to other channels are possible in MAGIC with the block transfer operations. Block transfers between general memory channels and display memory channels (W, X, Y, Z) are direct. Block transfers between two general memory channels must be done in two steps: From a specified general memory channel to a display memory channel and then from that display memory channel to the desired general memory channel. The W channel is usually used for this purpose. Partial channel block transfers occur from a specified sector address to the end of that channel (usually designated as "rev.", short for revolution pulse which occurs once per revolution of the drum and signifies the beginning or end of a channel). Single sector block transfers necessarily use the same sector in both channels involved. The instruction codes are as follows:

Octal Code	
00	BLOCK TRANSFER <u>to</u> W+X+Y+Z <u>from</u> general memory, specified sector to rev.
01	BLOCK TRANSFER <u>from</u> W+X+Y+Z <u>to</u> general memory, specified sector to rev.
02	Same as 00, except that only the specified sector is transferred.
03	Same as 01, except that only the specified sector is transferred.

In all cases, the source data of the block transfers remains unchanged. Operations 00 and 02 may specify any combination of W, X, Y or Z. The same can be done with operations 01 and 03, although there is as yet no known reason for doing so. The display may be cleared by block transferring a vacant general memory channel to X, Y and Z simultaneously.

Instruction example: Block transfer, sector to rev. starting with sector 63 of channel 434 to X and Y:

4350	0463

C. Jump Operations

A jump operation is one in which the contents of the IAC are replaced by the operand address portion of the IR, instead of the IAC proceeding in its normal counting sequence. This address must be limited to locations in general memory. There are some exceptions as to the jump address source. These will be apparent in the following text. A jump usually forces the program out of numerical sequence, and is used for reiterative loops, internal condition testing, sub-program linking, etc. The operation codes are as follows:

<u>Octal Code</u>

10 1. JUMP the IAC unconditionally to the operand address specified in the IR.

Instruction example: Jump to sector 77 of channel 44:

0441	0077

2. Sense Lights (see also op. code 17 below). If W, X, Y or Z is included with this instruction code, and if the corresponding sense light (SLW, SLX, SLY, SLZ) is ON, a jump will occur and the sense light (Fig. 7) will be turned off. If the sense light is off, a jump will not occur. An OR interrogation function is generated if more than one sense light is interrogated in the same instruction.

Instruction example: Jump to channel 044, sector 077 if SLX or SLY is on: | 0451 | 0477 |

Octal Code

12 JUMP the IAC to the operand address specified in the IR if the specified overflow flip-flop (W, X, Y or Z) is on. Any desired combination of W, X, Y or Z may be used. If this is done, any one of the specified overflow flip-flops being on will satisfy the condition for a jump. The overflow flip-flops are not reset.

Instruction example: Jump to sector 77 of channel 44 if X or Y overflow flip-flop is on: | 0451 | 2477 |.

14 JUMP the IAC to the sector address specified in the AAR bits $2^0 - 2^6$ if the specified A register (W, X, Y or Z) sees a zero in the sector specified in the IR. If more than one display channel (W, X, Y or Z) is used, a zero in any of them will satisfy the condition for a jump.

Instruction example: Jump to sector 77 of channel 44 if A_Z is zero in sector 50: | 0441 | 4250 | , AAR contents: X077. Note that the channel address is not affected.

Octal Code

16 JUMP the IAC to the operand address specified in the IR if the "NO HIT" flip-flop (NHFF) is on. The NHFF is not reset.

Instruction example: Jump to sector 77 of channel 44 if the NHFF is on: | 0441 | 6077 | . See section J (Scan Operations) for further information about the NO HIT flip-flop.

If the jump conditions for operations 10, 12, 14 or 16 are not satisfied, the jump will not occur and the next instruction is taken in normal sequence. The source data (except for the sense lights) for any jump operation remains unchanged regardless of the success of the jump.

D. Control Operations

This class of operations includes many functions which will be discussed separately below.

1. Interrupt: When an interrupt instruction occurs in the IR, the machine remains in phase two until one of the interrupt controls is depressed. At that time an interrupt sector address unique to the control used is placed in the $2^0 - 2^6$ bit positions of the IAR. These IAR bits are then transferred to the sector portion of the IAC and the interrupt operation is terminated. As a result, the processor jumps to the sector so specified. The channel address in the IAC and the contents of the IAR are unchanged. See section VI-B3e for a more detailed description of the interrupt controls. The interrupt operation codes are as follows:

<u>Octal Code</u>

11 INTERRUPT: Hold the machine in phase two until an interrupt control is depressed. Then modify the IAC sector address as described above and terminate the operation.

Instruction example: Perform an Interrupt: |XXX1|1XXX|.

This instruction is usually accompanied by a list of unconditional jump instructions (in the same channel) in the sectors addressed by the interrupt controls for branching to the desired subroutines. See Fig. 10.

2. Fill control for display B registers (X, Y and Z):

<u>Octal Code</u>

13 TRANSFER to B_x and B_y registers from locator coordinate registers and transfer to B_z from the keyboard register and the Z controls. No other data or registers are affected.

Instruction example: Perform a transfer operation: |XXX1|3XXX|.

3. External Control:

<u>Octal Code</u>

15 Output a single t_{13} pulse to an external device. Nothing within the processor itself is effected. At present, four devices may be controlled; one each for W, X, Y or Z. Obviously, up to four devices may be controlled simultaneously.

Instruction example: Enable an external device: |XXX1|5XXX|.

4. Halt, Breakpoint, and Sense Light Control: The machine language codes are listed below for the various operations included in this instruction.

Octal Code	Sub-code W X Y Z	Function	Instruction
17	1 1 1 1	HALT	XX31 76XX
	1 0 0 0	Set SLW	XX21 70XX
	0 1 0 0	Set SLX	XX11 70XX
	0 0 1 0	Set SLY	XX01 74XX
	0 0 0 1	Set SLZ	XX01 72XX
	0 0 0 0	BREAKPOINT	XX01 70XX

1. HALT: The machine will halt upon receipt of this instruction, regardless of the operation mode (see Section VI-B1g) it is in.
2. BREAKPOINT: Halts the machine only if it is in breakpoint operation mode.
3. Sense Light Set: Turns <u>on</u> the sense lights specified by W, X, Y or Z. Although more than one sense light may be turned on with the same instruction, obviously all cannot be turned on at once. Such an instruction would be a halt instruction (see also op. code 10). This class of instructions does not affect the machine in any way other than that which is specified above.

E. Fill Operations

This class of operations will fill the following registers from the address of general memory specified in the instruction: AAR, MR, IOR, IAR. If W or X or Y or Z or any combination thereof is included in the instruction, the following registers may be filled: B_w, C_w, B_x, C_x, B_y, C_y, B_z and C_z. The data in the specified memory address in all cases remains unchanged. The instruction codes are as follows:

Octal Code

21	FILL B (W+X+Y+Z) register.
22	FILL C (W+X+Y+Z) register.
23	FILL the alternate address register (AAR).
24	FILL the mask register (MR).
25	FILL the I/O register (IOR).
26	FILL the interrupt address register (IAR).

When using operations 21 or 22, any combination of W, X, Y or Z may be specified.

Instruction example: Fill C_x, C_y and C_z registers from channel 334, sector 105: | 3352 | 2705 | .

F. Empty Operations

The same registers as listed in the Fill operations may be emptied into the address of general memory specified in the instruction. The contents of all registers except the B registers and the IOR remain unchanged. These registers are reset to zero as they are emptied. The instructions codes are as follows:

Octal Code

31	EMPTY B (W+X+Y+Z) register.
32	EMPTY C (W+X+Y+Z) register.
33	EMPTY alternate address register (AAR).
34	EMPTY mask register (MR).
35	EMPTY I/O register (IOR).
36	EMPTY interrupt address register (IAR).

When using operations 31 or 32, any combination of W, X, Y or Z may be specified.

Instruction example: Empty the mask register into channel 334, sector 105: | 3343 | 4105 | .

G. Arithmetic Operations

It may be recalled that data within MAGIC is unsigned, and all numbers are assumed positive. There is a binary full adder in each subordinate list processor. Therefore, up to four additions may be performed simultaneously (i.e., any combination of W, X, Y and Z may be used.) The contents of the C register are added to the contents of the A register, with the result being written back on the drum at the same sector address. Subtraction is performed by applying the complement of the C register to the adder. When "subtracting" in this manner, the C register output to the adder is in the form of a ones complement. Thus for "negative" numbers, 7777_8, regarded as negative, is the same as 0, $7776_8 = -1$, etc., and 0000_8, regarded as a "negative" number, is the same as $-(7777)_8$ or $-(4095)_{10}$. In this method of subtraction the answer will always have to be corrected, either by adding one or by recomplementing. The occurrence of an overflow indicates that a one should be added to the result, and that it is positive. If there is no overflow it indicates that the result is "negative"; it may be left as is or recomplemented as desired.

Modulo 2 addition is performed by simply not permitting the propagation of any carries during the addition. Since the adders are part of the display processors, the channel address of the instruction has no meaning. The sector address of the instruction specifies the sector within the particular display channel to be added to its C register, or the starting sector for addition from that sector to rev.

The contents of the C register involved remain unchanged at the completion of the addition operation. If an overflow occurs, the respective overflow flip-flop is set. Also, the respective overflow flip-flop is reset at the beginning of the arithmetic operation. The instruction codes are as follows:

Octal Code	
40	ADD A and C (W+X+Y+Z), specified sector.
41	ADD A and $\bar{C}$ (W+X+Y+Z), specified sector.
42	ADD A and C (W+X+Y+Z), Mod. 2, specified sector.
43	ADD A and $\bar{C}$ (W+X+Y+Z), Mod. 2, specified sector.
44	Same as 40, specified sector to rev.
45	Same as 41, specified sector to rev.
46	Same as 42, specified sector to rev.
47	Same as 43, specified sector to rev.

Instruction example: "Subtract" 2_{10} from sector 27 of display channels X and Y: |XX14 |1427|, $C_{x,y}$ contents: 0002

or: |XX14 |0427|, $C_{x,y}$ contents: 7775

The X and Y overflow flip-flops, having been reset at the beginning of the operation, would be set at the end of the operation in either of the above examples. During a 44-47 operation, any sector addition between the specified sector and rev. that results in an overflow will set the respective overflow flip-flop.

H. Insert Operations

The insert operations, applicable only to W, X, Y and Z are true inserts. That is, a data word residing in the B register of the display processor involved will be written into the specified sector of its channel and the previous contents of that sector will be moved to the next sector and so on to the end of the channel. Data appearing before the specified sector is unaffected. The contents of the last word (word 127_{10}) in the channel previous to the insert operation will be in its B register at the end of the insert operation.

As implied above, a necessary prerequisite for an insert operation is the filling of the respective B register. Such may be performed via program or via the keyboard (when working with the display). An insert may be performed in any desired combination of the W, X, Y or Z channels. The instruction codes are as follows:

Octal Code

50 INSERT the contents of B (W+X+Y+Z) register into its respective memory channel at the sector address specified in the instruction.

Instruction example: Insert B_W contents into sector 30 of W: | XX25 | 0030 |

W contents before insert:

address:	29	30	31	32
contents:	a	b	c	d

W contents after insert:

address:	29	30	31	32
contents:	a	B_W	b	c

Octal Code

51 INSERT the contents of B (W+X+Y+Z) register at the first zero word detected in its respective display channel specified in the instruction. The zero word detected is retained and appears in the first word following the inserted word at the end of the operation.

Instruction example: Insert B_W contents into 1st zero detected in channel W: | XX25 | 1XXX |

W contents before insert:

address:	10	11	12	13
contents:	a	0^1	b	c

1: first zero in channel.

W contents after insert:

address:	10	11	12	13
contents:	a	B_W	0^2	b

2: location of new first zero in channel.

It may be noted that any or all display channels may be designated in this operation. The first zero detected in any of the designated channels will result in an insert at that sector for all of the designated channels.

I. Delete Operations

The delete instructions, applicable only to W, X, Y and Z, are true deletes. That is, a data word residing in a display channel sector specified in the instruction is deleted and all following data words in the channel move up to fill the gap (see instruction examples below). The deleted word is lost. Data appearing before the specified sector is unaffected. In keeping with a true delete, the last word in the channel is zeroed after its contents have been moved up to the next to the last word. The contents of the last word are therefore not lost. A delete operation may be performed in any desired combination of W, X, Y or Z. The instruction codes follow:

Octal Code

52 DELETE the contents of W, X, Y or Z channel(s) at the sector address specified in the instruction.

Instruction example: Delete sector 33 of X, Y and Z: | XX15 | 2633 |

The following holds also for Y and Z:

X contents before delete:

address:	32	33	34	35
contents:	a	b	c	d

X contents after delete:

address:	32	33	34	35
contents:	a	c	d	e

Octal Code

53 DELETE the contents of W, X, Y or Z channel(s) at the first zero word detected in the channel(s).

Instruction example: Delete 1st zero found in channel W:

XX25	3000

W contents before delete:

address:	32	33	34	35
contents:	a	0*	b	c

*first zero in channel

W contents after delete:

address:	32	33	34	35
contents:	a	b	c	d

It may be noted that any or all display channels may be designated in this operation. The first zero detected in any of the designated channels will result in a delete at that sector for all of the designated channels.

J. Masked Scan Instructions

This class of instructions applies only to channels W, X, Y and Z. The scan operations cannot be performed on data in general memory. During one revolution of the drum, the execution of a scan operation will compare the data words in the specified display channel with the contents of the C register associated with that channel. All scan operations will scan data sequentially from the sector specified in the instruction to the end of the channel. An exception: the COMPARE operation (see below) which operates only on the sector specified in the instruction.

Since the scan operations are masked, the masking data should be loaded into the mask register (MR) prior to the execution of a scan operation. Masking is performed on a serial basis. Thus a "1" in any bit position in the mask register will cause that bit position of the A and C registers involved to be compared. Therefore, the "ones" existing in the mask register at the time of the scan determine the bit field scanned within each data word. For example: to scan for the first whole word in channel X which is less than a number in C_x one would load the MR with 7777_8. To find the first least significant octal digit in channel X which is less than the least significant octal digit in C_x one would load the MR with 0007_8.

There are two direct ways to scan for a word which contains all zeros: (1) The A = 0 scan operation itself, and (2) the A = C operation in which all zeros are placed in the C register involved and 7777_8 is placed in the MR. The A = 0 operation is not masked, since it makes use of the zero detectors tied to the A registers (see Fig. 4). Therefore, in order to scan a bit field less than a full word for all zeros within that field, one must use the A = C operation.

A scan operation may be either successful or unsuccessful. If the scan operation is successful, the following events occur: (1) The address of the successfully scanned word is transferred from the MWC to bits $2^0 - 2^6$ of the AAR. The remaining AAR bits are unaffected. (2) The successfully scanned whole word is placed in the B register associated with that channel regardless of the mask field. (3) The scan operation is terminated at the time of this "hit". If the scan is unsuccessful, the following occurs: (1) The scan operation self-terminates at the end of the channel. (2) The "NO HIT" flip-flop (NHFF) is turned on, lighting the NHFF indicator on the processor control panel (the NHFF is reset at the beginning of every scan operation). (3) The AAR contents remain unchanged. (4) The last word (word 127_{10}) of the scanned channel is placed in the respective B register.

The scan operations may be performed on any combination of W, X, Y or Z simultaneously. If this is done, the following characteristics should be noted. The first word that satisfies the scan requirements in any one of the channels being simultaneously scanned will terminate the scan. The sector address of this first word will appear in the AAR as previously described. At the end of such a scan each B register involved will contain the data in that sector of their respective channels. If none of the scanned channels contain data that will satisfy the scan, the operation terminates in the same manner as for a single channel scan. Also, the masking data is necessarily the same for simultaneously scanned display channels.

Any scan operation will not change data within the scanned display channels or the data contained within the respective C registers. The instruction codes are listed below:

Octal Code	
60	SCAN (Masked) for A = C (W+X+Y+Z)
61	SCAN (Masked) for A > C (W+X+Y+Z)
62	SCAN (Masked) for A < C (W+X+Y+Z)
63	SCAN (Not Masked) for A = 0 (W+X+Y+Z)
64	SCAN (Masked) for A ≠ C (W+X+Y+Z)
65	SCAN (Masked) for A = C in both X and Y simultaneously
66	COMPARE (Masked) A and C (W+X+Y+Z) for A = C within a specified sector.

Instruction examples: 1. Scan channel W for the first word after sector 40_8 whose second most significant digit is less than 5: | XX26 | 2041 |. These registers must be loaded prior to the scan operation: C_W contents: $X5XX_8$

MR contents: 0700_8

2. Using the COMPARE instruction, check to see if the least significant bit of word 170_8 in W is a "ONE": | XX26 | 6170 |.

These registers must be loaded prior to the scan operation:

C_w contents: XXX1

L.S.B.

MR contents: 0001

It should be noted here that scan instruction 65 requires a simultaneous hit in both X and Y to satisfy the scan requirements. This is useful when scanning for a desired coordinate. It is not necessary for the C_x and C_y registers to be loaded with the same data. As previously stated, the words scanned in X and Y will be identically masked. This operation applies only to X and Y which must be included in the instruction.

K. Shift Operations

The shift operations pertain only to W, X, Y and Z, and apply to the shifting of the bits (right or left) within a single data word. However, as seen below, all the words between a specified sector and rev. may be shifted in one operation. Overflow during the shift operations is not propagated to the adjacent data words and is lost.

Shifting "right" decreases the binary weights of the shifted bits and shifting "left" increases the binary weights of the shifted bits. (See the examples below). All data outside of the field of shift is not affected. Also no registers are affected by this operation. A zero is inserted "behind" the shift in all cases. For example: if the word 0002_8 is shifted left, the resulting word is 0004_8, not 0005_8. The execution of the shift operation shifts the designated word(s) one bit position only. Thus, to shift two bit positions, the shift operation must be performed twice. The shift operation may be performed simultaneously on any combination of channels W, X, Y or Z. as desired. The instruction codes are listed below.

Octal Code

70 SHIFT RIGHT (W+X+Y+Z) the contents of only the sector specified in the instruction.

71 SHIFT RIGHT (W+X+Y+Z) from the sector specified in the instruction to the end of the channel.

72 SHIFT LEFT (W+X+Y+Z) the contents of only the sector specified in the instruction.

73 SHIFT LEFT (W+X+Y+Z) from the sector specified in the instruction to the end of the channel.

Instruction examples: 1. Shift right sector 10_8 of channel W one bit: | XX27 | 0010 |

Contents of sector 10_8 of W before shift: 7043_8

Contents of sector 10_8 of W after shift: 3421_8

2. Shift all words in channel W left one bit: | XX27 | 3000 |

Contents of (for example) sector 20_8 of W before shift: 7043_8,

Contents of sector 20_8 of W after shift: 6106_8

VI. MANUAL CONTROLS

MAGIC is divided into three sections: (1) primary display with light pen (right display console), (2) secondary display (left console), and (3) control console (center). This section will describe the functions of the manual controls on the consoles listed above which are available to the operator. A schematic rendition of the physical layout of MAGIC appears in Fig. 5, showing the location of the various control panels.

A. Display Console Operation

The display consoles each consist of a CRT with magnetic deflection amplifiers, preamplifiers, power supplies and display controls. The CRT deflection scheme is a dual one; consisting of a main deflection yoke, and a "piggy-back" yoke for relatively fast, short excursion deflections. The two deflection systems are electronically independent.

1. Power

The power controls for both display units are independent of the processor. The power switches are located at the bottom left corner of the front of the secondary display chassis. BEFORE TURNING DISPLAY POWER ON OR OFF IT IS VERY IMPORTANT THAT THE BRIGHTNESS CONTROLS OF EACH DISPLAY BE TURNED TO MINIMUM BRIGHTNESS OR FULL COUNTER-CLOCKWISE POSITION. Failure to do this may cause the CRT tube faces to be burned. To turn display power on: (1) turn on the "FIL" switch, wait one minute, then (2) turn on the "H.V." switch. To turn power off, turn off both switches simultaneously. Do NOT turn off the "FIL" switch before the "H.V." switch.

2. CRT Adjustments

Both displays have focus and brightness controls located near each CRT face. The brightness controls should be adjusted so that all non-intensified display data and CRT retraces are invisible.

3. Deflection Amplifier Adjustments

The displays have identical amplifiers and associated controls for the main and piggy-back (labeled AUX.) deflection systems. The amplifier controls are located as shown in Fig. 5. Gain and balance controls for the X and Y axes are identical. The gain controls are self explanatory,

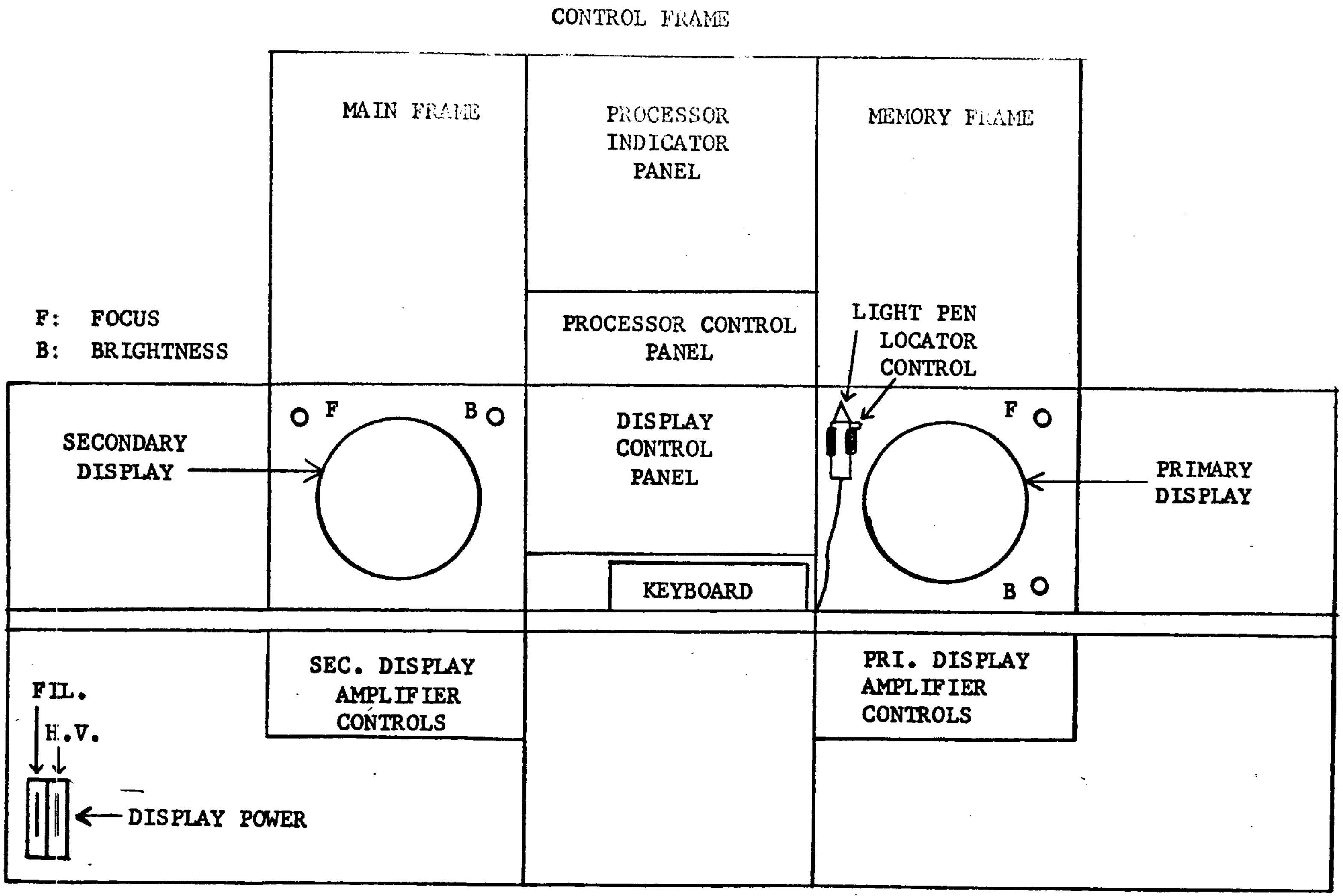


Figure 5. MAGIC Physical Layout

and the balance controls effect the centering of the display. Amplifier saturation may occur if the balance controls are positioned in the vicinity of their control limits. Adjustment of the piggy-back balance controls will affect the centering of the entire display.

B. Control Console Operation

The control console is divided into three sections: (1) processor control panel, (2) processor indicator panel, and (3) display control panel.

1. Processor Control Panel (see Fig. 6)

a. POWER ON/OFF: This alternate action pushbutton controls power to the processor. When turning power on, hold the master reset switch in until the power supplies get up to their operating voltage (a few seconds). It is also safe practice to hold the master reset switch in as power is turned off. The processor is ready to operate when the drum is up to speed. However, as the drum turns up to speed, the processor may go into sporadic operation. This may be halted by depressing the phase one control.

b. MASTER RESET (labeled MR): This pushbutton resets the processor. All registers displayed on the processor indicator panel are reset, as well as the sense lights, the run/halt control and various other logic functions pertinent to a reset condition within the processor.

c. RUN: Depression of this pushbutton starts the processor. If the processor is in phase one, the first instruction will automatically be fetched from the address specified in the IAC. If the processor is in phase two, the instruction in the IR will be performed first.

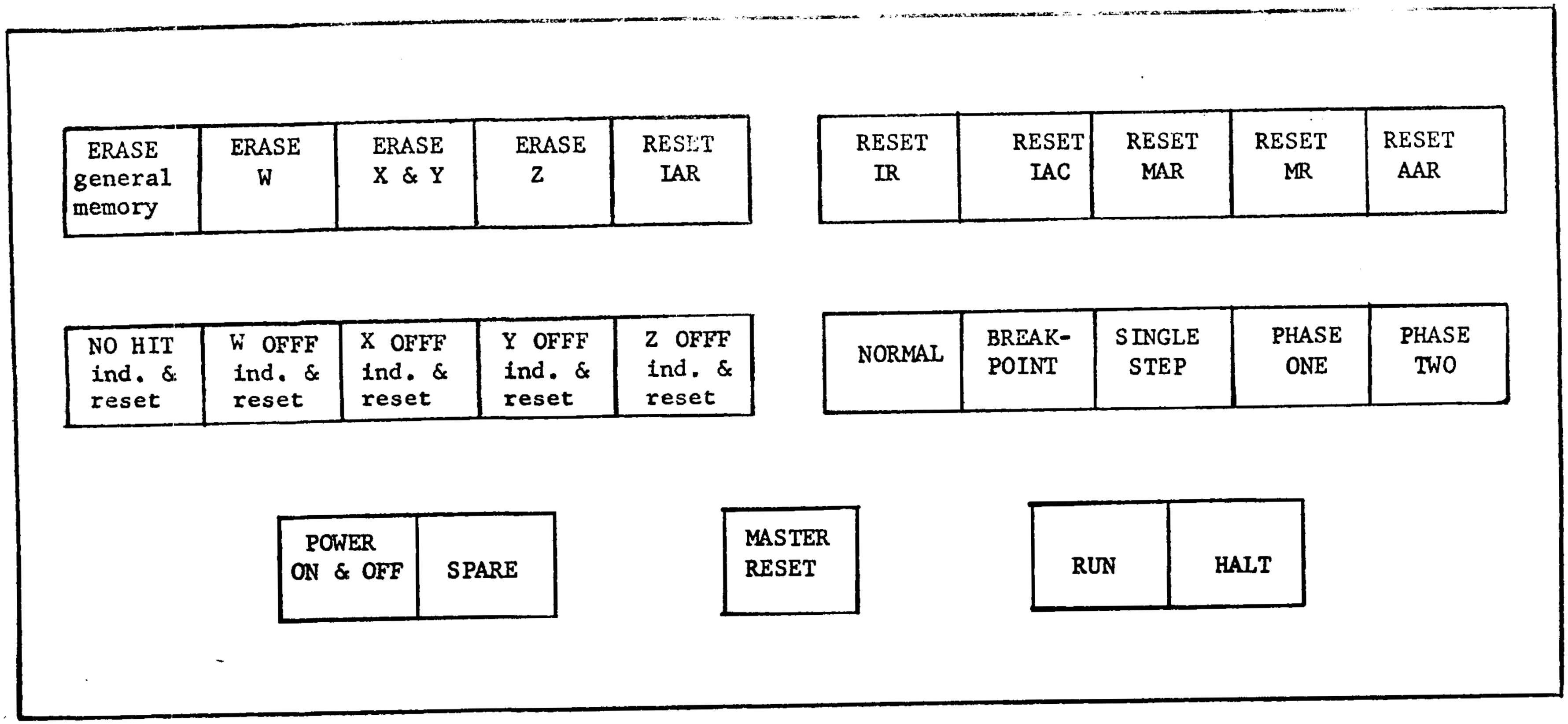


Figure 6. Processor Control Panel

d. HALT: Depression of this pushbutton causes the processor to halt at the end of the current phase one cycle. The machine then halts with the current instruction in the IR, the current instruction address in the MAR, the address of the next instruction in the IAC, and the phase control in phase two mode. Therefore, the first machine function to be performed when the RUN control is next depressed is the phase two operation displayed in the IR.

e. PHASE ONE control: When the processor is halted, depression of this pushbutton forces the processor to begin in phase one, fetching the first instruction from the address shown in the IAC.

f. PHASE TWO control: When the processor is halted, depression of this pushbutton forces the processor to begin in phase two, performing the operation shown in the IR.

g. Operation mode controls: There are three modes of operation in MAGIC: "Single Step", "Breakpoint", and "Normal". If the processor is in the single step mode, one phase one/phase two cycle will be performed each time the RUN pushbutton is depressed. The processor will halt in phase one. If the machine is in breakpoint mode, the performance of a breakpoint or halt operation will halt the processor in phase one. If the processor is in normal mode, it will continue to operate until a halt instruction is executed, at which time the processor will halt in phase one. The mode controls are not affected by master reset. The operation mode may be changed at any time. An accepted method of manually halting the processor when running in normal or breakpoint mode is to depress the single step pushbutton.

h. Individual register RESET controls: As shown in Fig. 6, the registers displayed on the processor indicator panel may be individually reset by their respective reset controls. These are: IR, MAR, IAC, MR, AAR and IAR. The one exception is the IOR, whose reset control is located in the lower right-hand corner of the processor indicator panel (see Fig. 7). It should be noted here that the IAC is always reset to channel address 700_8. The purpose of this is discussed in VII-A2.

i. ERASE controls (see also Section VI-B-1i). One may erase any memory channel via the erase controls. Erasure of the display channels may be done at any time. The general memory channels (including the I/O, X', Y' and Z' channels) may be erased by a procedure outlined in Section VII-A5.

j. Overflow indicator/controls: Within each display processor (W, X, Y and Z) there is associated with the arithmetic unit an overflow flip-flop (OFFF). When an OFFF is set, its respective indicator/control will light. Depression of these indicator/controls reset the respective OFFF. See the discussions of the arithmetic and jump instructions for further information on the functions of the OFFFs.

k. NO HIT indicator/control: The indicator portion of this control, when on, indicates that a scan operation just performed was unsuccessful (i.e. "no hit"). Depression of this control manually resets the no-hit flip-flop (NHFF). See the discussions of the jump and scan instructions for details of the functions of the NHFF.

2. Processor Indicator Panel (see Fig. 7)

This panel displays via indicator lamps the contents of the following registers: IR, IAC, MAR, AAR, MR, IAR and IOR. Pushbutton controls for setting up the MAR and IOR bit-by-bit are provided beneath

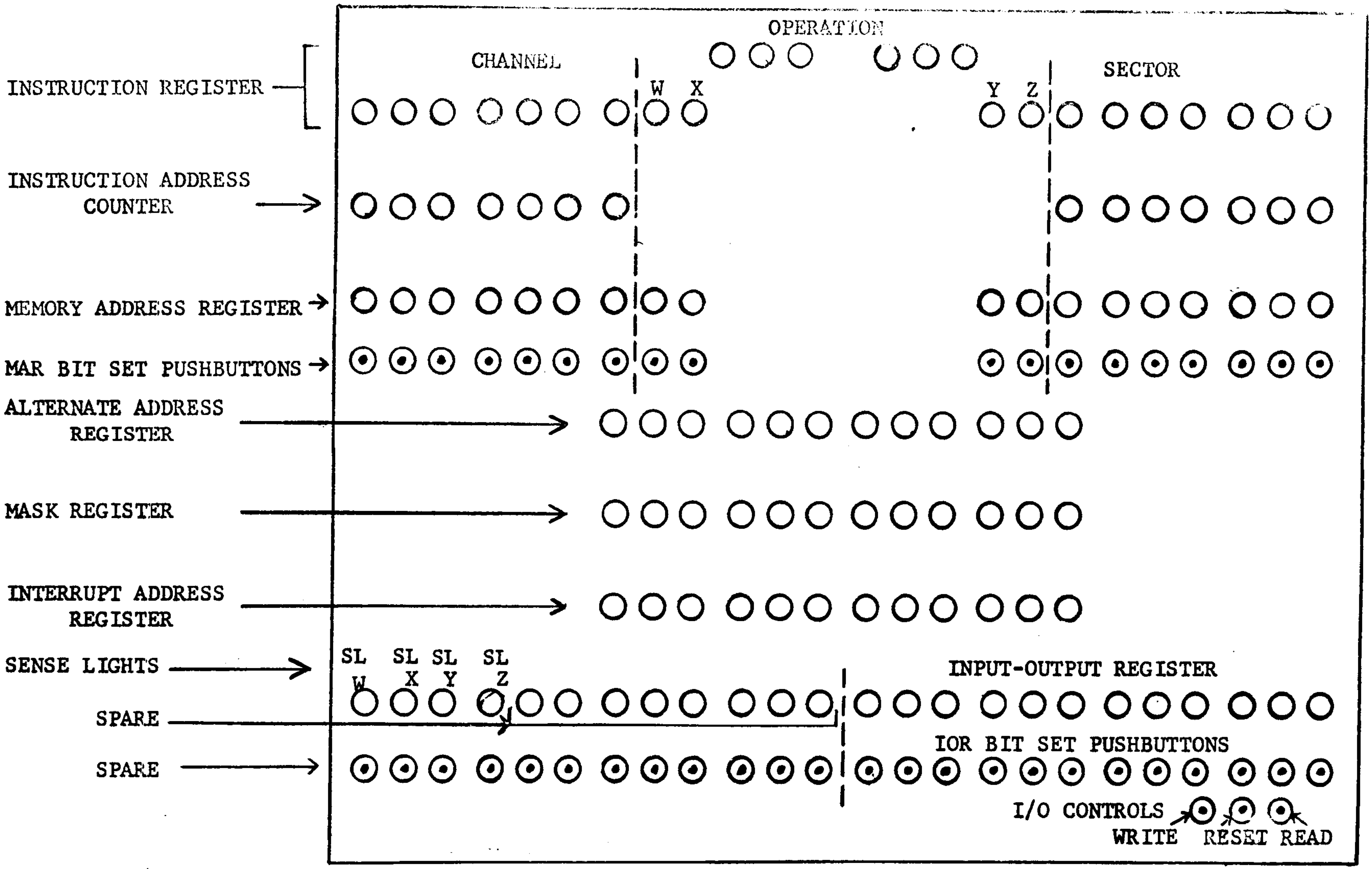


Figure 7. Processor Indicator Panel

the indicators for these registers. The manual controls for the IOR and its associated I/O channel are located in the lower right corner of the panel. These are "READ" (I/O memory channel to IOR), "RESET" (IOR), and "WRITE" (IOR to I/O memory channel). The I/O process is discussed in Section VII-A2. The sense lights are also displayed and may be manually reset by the Master Reset Control.

3. Display Control Panel (See Fig. 8)

The display control panel is divided into the following sections in the discussion below: (a) I/O channel sector selector, (b) character generator control, (c) "Z" controls and keyboard, (d) light pen and locator controls, (e) interrupt controls.

a. I/O channel sector selector: This three unit thumbwheel selector determines the sector of the I/O channel to be used in conjunction with the external control of this channel. It may be operated at any time.

b. Character generator control: This alternate action switch links the character generator to the primary or secondary display. Thus, one display unit is free to display a mixture of alphanumeric characters and other data while the other display unit is limited to displaying non-alphanumeric data only. This does not affect the data stored in the Z or Z' channels.

c. "Z" controls and keyboard. The Z display channel contains control data for the various display presentation parameters. The binary code for alphanumeric characters is included. The Z control word format is shown in Fig. 9.

The row of toggle switches (Z controls) at the bottom of the display control panel control the display presentation parameters and are so labeled. Depression of a toggle switch turns it "on", raising the switch turns it "off". Depressing two or more switches for the same entry within a given parameter (e.g. brightness) is not allowable.

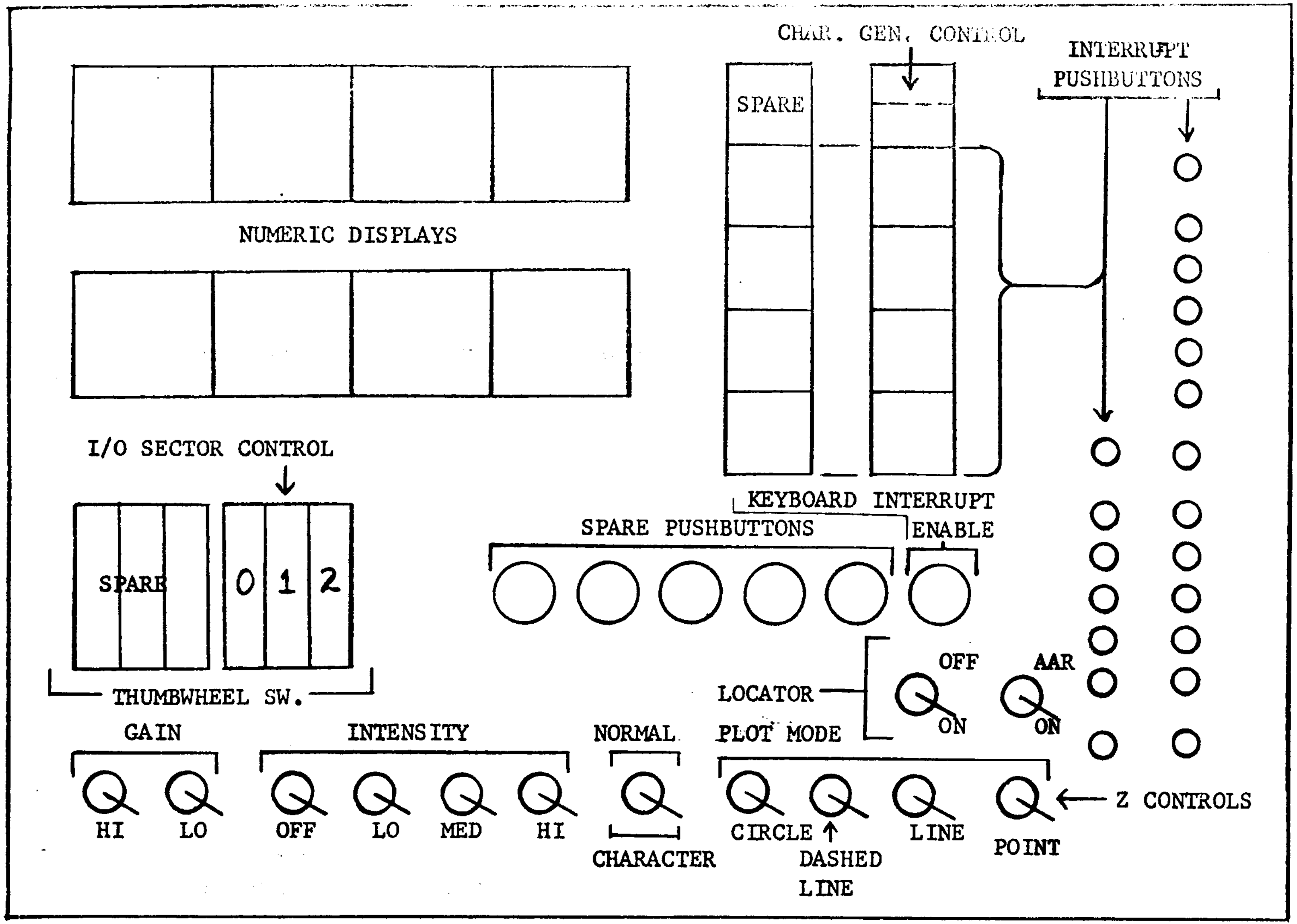


Figure 8. Display Control Panel

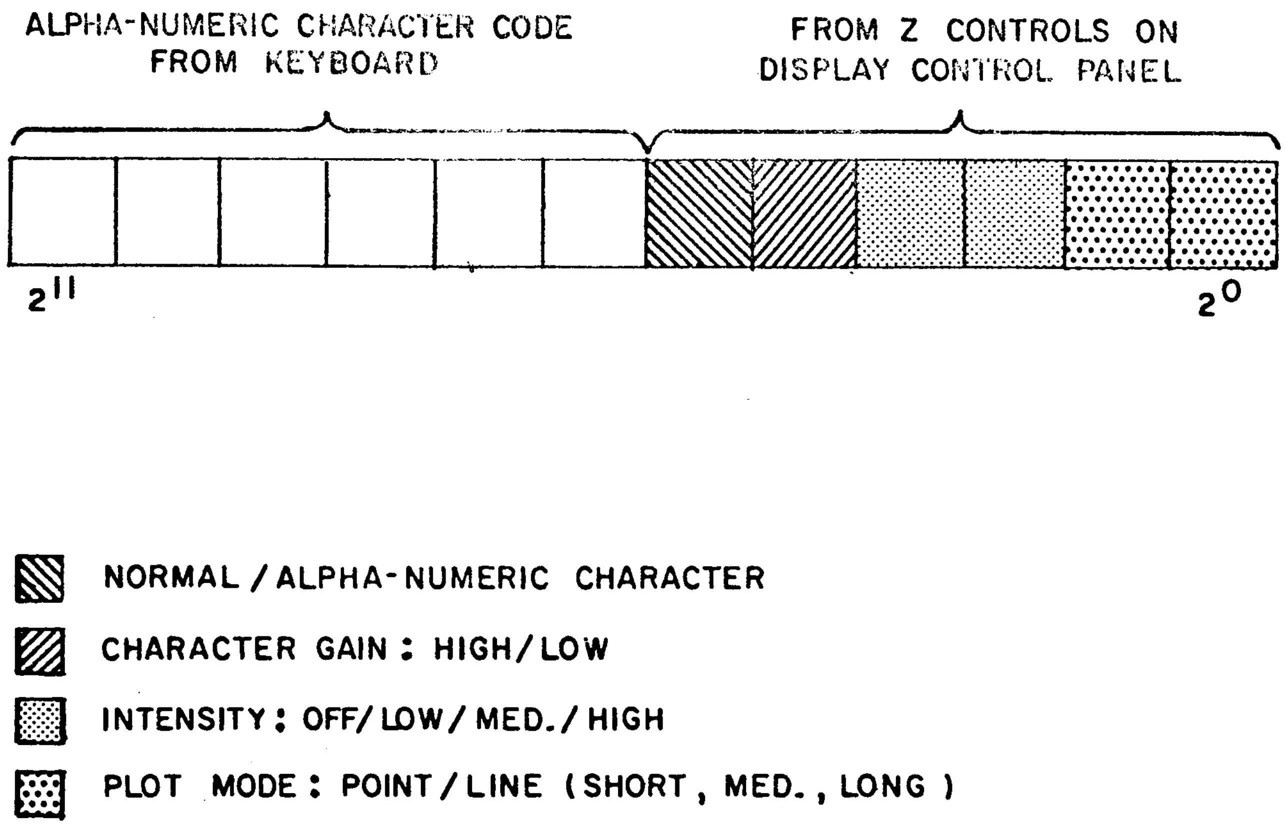


FIGURE 9. Z WORD FORMAT

The keyboard enters appropriate alphanumeric data codes from itself and the display presentation codes from the Z controls into the B_z register upon depression of any key. This also loads the locator coordinates (see below) into the B_x and B_y registers. A subsequent insert operation can then load this X, Y and Z data into the display memory, completely describing the presentation of a display subject. The character generator in MAGIC is capable of generating only alphanumeric characters. Thus, depression of keyboard keys representing other symbols (see Appendix C) will result in a space (no character) being displayed. The keyboard may operate as an interrupt by depressing the keyboard interrupt enable pushbutton (Fig. 8). When the processor is waiting for an interrupt (operation 11), depression of any key will terminate the interrupt operation. The B_x, B_y and B_z registers are loaded as described above. The keyboard power switch should be operated only when the processor power is off. The four function keys at the left of the keyboard are additional interrupt controls.

d. Light pen and locator controls: The method used in MAGIC to generate X-Y coordinates of a display incorporates a cross-hair or "locator" generated on the primary display. The locator switch on the display control panel turns the locator on or off. By placing the light pen associated with the primary display over the locator and depressing the locator movement control switch on the light pen body, one may "drag" the locator about the tube face, since the locator is designed to center on the light pen. The coordinates of the locator center may be transferred to the B_x and B_y registers for subsequent insertion by depressing a key on the keyboard or by performing the TRANSFER operation.

For identification of a point, line or alphanumeric character one need only to place the light pen over the desired feature (locator may be off) and momentarily depress the "AAR" toggle switch. This enters the sector address of the corresponding data word from the MWC into bits $2^0 - 2^6$ of the AAR.

e. Interrupt controls: A powerful feature in MAGIC is the ability to perform programmed interrupts. The interrupt jump instruction (see section V-D1) in effect asks the operator what he would like to do next. By depressing an interrupt pushbutton (see Fig. 8) after the IR receives an interrupt instruction, the sector address of the next instruction (in the IAC) to be fetched is changed to the sector address which is unique to the interrupt control depressed. This new address usually contains an unconditional jump instruction to a subroutine used for manipulation of display data. Figure 10 shows a flow chart of such a programmed interrupt. The same interrupt channel arrangement may be repeated in as many channels as desired. This permits the interrupt controls to be "redefined" for as many functions or groups of functions as desired. Thus, a particular interrupt control may represent a picture rotation subroutine to one operator or a multiplication subroutine to another operator. Additional interrupt controls not shown in Fig. 8 consist of the four function keys at the extreme left of the keyboard and the "floating" interrupt control panel in front of the primary display.

From the preceding paragraphs it is seen that the light pen, locator, Z and interrupt controls, along with the keyboard operate as a team for generation and manipulation of display data. A detailed sequence of operation for these display controls will be listed in the following section.

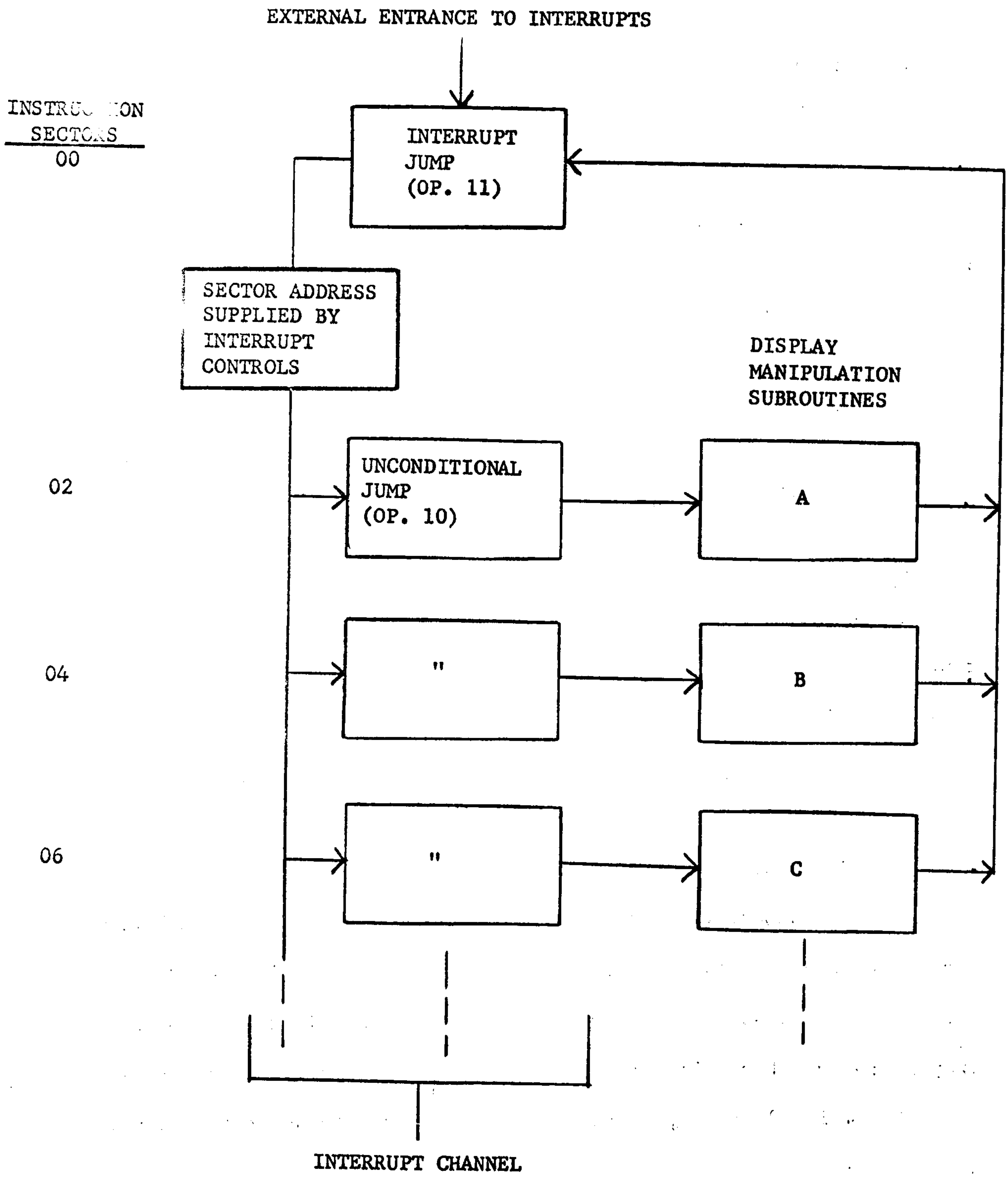


Figure 10. Interrupt Programming Flow Diagram

VII. MACHINE OPERATION

A. Processor

The functions of the manual processor controls on the processor control panel are described in section VI-B.

1. Processor Power

The following procedure should be followed when turning on the processor:

a. Depress and hold master reset.

b. Depress power on/off pushbutton.

c. Continue to hold down master reset.

d. As the drum turns up to speed, the processor may start to operate erratically. If this happens, momentarily depress the phase one control and the single step control.

e. Master reset may be released when the drum is at or near operating speed.

The following procedure should be followed when turning off processor power:

a. Depress and hold master reset.

b. Depress power on/off pushbutton.

c. Hold master reset down for a few seconds after power has been turned off.

2. I/O Control

The I/O channel (channel 700_8) may be addressed directly with the I/O channel controls on the processor display panel and display control panel. All programs can be entered a single word at a time via this channel. If a program to be entered is written for some other general memory channel, the first three instructions, starting in sector zero, to be entered should be as follows:

1) A whole channel block transfer from channel 700_8 to W.

2) A whole channel block transfer from W to the channel for which the program was written.

3) An unconditional jump to the starting address of the program.

This is necessitated by the fact that, as may be observed, the IAC is reset to sector zero of channel 700_8. The first instruction will always be read from this address. Data entry or retrieval from the I/O channel may be accomplished at any time, even while the processor is running. Data entry may be performed as follows:

a. Depress master reset.

b. Set up the desired sector address with the I/O sector selector on the display control panel.

c. Set the desired bits of the IOR by depressing the IOR bit set pushbuttons on the processor display panel.

d. Depress the I/O "write" pushbutton and repeat steps 2-4 for entry of remaining data.

Mistakes in setting up the IOR may be corrected by depressing the IOR reset pushbutton and re-entering the data. If a sector has already been filled with erroneous data, simply perform steps b-d above.

Reading data from channel 700_8 into the IOR may be accomplished as follows:

a. Select the desired sector with the thumbwheel switches.

b. Depress the I/O "READ" pushbutton.

3. RUN/HALT Procedures

a. Starting the processor

1) If master reset has been performed:

a) Depress PHASE ONE pushbutton unless the phase one light is already on.

b) Depress desired mode of operation (manual, breakpoint or single step).

c) Depress RUN pushbutton.

2) If master reset has not been performed (such as when operating a program in single step mode):

a) Depress PHASE ONE pushbutton unless the phase one indicator is already on.

b) Depress RUN pushbutton.

The processor may be started in phase two for debugging purposes. First, the operation shown in the IR will be performed on the address shown in the MAR. Then the next instruction will be taken from the address shown in the IAC prior to the depression of the RUN pushbutton unless the operation shown in the IR calls for an IAC address modification.

b. Manually halting the processor

The processor may be manually halted if it is in normal or breakpoint mode by:

1) Depressing the SINGLE STEP pushbutton. The processor will halt in phase one, ready to fetch the next instruction from the general memory address shown in the IAC when the RUN pushbutton is depressed.

2) Depressing the HALT pushbutton. The processor will halt in phase two, ready to perform the instruction shown in the IR on the general memory address shown in the MAR. The master reset pushbutton should never be used to halt the processor except in a case of emergency.

4. MAR Control

The MAR may be set to any channel and/or sector address manually by depressing master or MAR reset and then depressing the desired MAR bit set pushbuttons. This should be done only when the machine is halted.

5. Memory Channel Erasure

Erasure of a display channel may be accomplished at any time by depressing the appropriate erase controls. (See Fig. 6). Erasure of a general memory channel may be accomplished by:

a) Placing the processor in halt mode.

b) Depressing master reset or MAR reset.

c) Selecting the desired channel to be erased using the MAR bit set pushbuttons.

d) Depressing the general memory erase control.

Erasure of a channel sets the entire channel to an "all zeros" state. General memory erasure cannot be accomplished when the processor is in run mode.

B. Primary Display

The functions of the various display controls are described in section VI. This section deals with the use of the primary display. The operator is basically concerned with two displays manipulations:

1) Using the light pen and locator to generate display data.

2) Using the light pen and "AAR" switch to identify existing display data.

The control operation sequences for these two basic display manipulations are described below.

Some basic characteristics should be stated here before proceeding to the operation sequences. (1) The locator may be turned on or off and may be moved about with the light pen at any time. (2) The keyboard keys and Z controls may be manipulated at any time provided that the operator is aware of the existence and possible interference of any TRANSFER instructions in his program.

1. Display control sequence for generation of display data via light pen.

   a) Turn locator on.

   b) Move locator to desired position with the light pen. The locator movement control on the light pen must be depressed to move the locator.

   c) Select desired display characteristics with the Z controls. This may actually be done prior to a or b above.

   d) Depress desired keyboard key if keyboard is being used to perform the transfer operation.

   e) Depress the interrupt control that has been previously designated by the operator to insert display data into the display data list in channels X, Y and Z, unless the keyboard is being used as an interrupt control (see section VI-B3c).

   f) Repeat steps a-e as desired.

2. Display control sequence for identifying existing display data.

This sequence determines the memory address of the display subject at which the light pen is pointed. The address appears in the AAR. Alternate addressing is then used for manipulation routines concerning the identified subject. The control sequence is as follows:

a) Turn locator off.

b) Position light pen over subject to be identified. The locator movement control on the light pen should not be depressed.

c) Momentarily depress "AAR" switch.

d) Depress the interrupt control that has been previously designated by the operator to perform desired manipulations (such as delete) upon the identified subject.

e) Repeat steps a-d as desired.

ADDENDA

The following is a list of miscellaneous hardware and software modifications incorporated in MAGIC since the text of this report was finalized. With this addenda, the operational characterisitcs of MAGIC are current as of April 1, 1965.

A. Operation Code 15: The function of this operation code has been changed from EXTERNAL CONTROL to MISC. REGISTER OPERATIONS.

OP. CODE	SUB-CODE	FUNCTION	INSTRUCTION
15	X	SHIFT RT. MR 1 BIT.	XX11 50XX
"	Y	JUMP IF AAR BITS 20 - 26 = ALL 1'S.	XX01 54XX
"	Z	INCREMENT AAR.	XX01 52XX

Any combination of X, Y or Z may be used. In the case of the combination of Y and Z, the increment function will be performed first. The operation code 15 plus sub-code Y will jump to the operand address specified in the instruction if the jump conditions are met.

B. The current valid general memory channel addresses are as follows: 000-260, 374-664, 700 (octal notation). Channels with special characteristics are as follows: 000: all zeros (no read-write head)

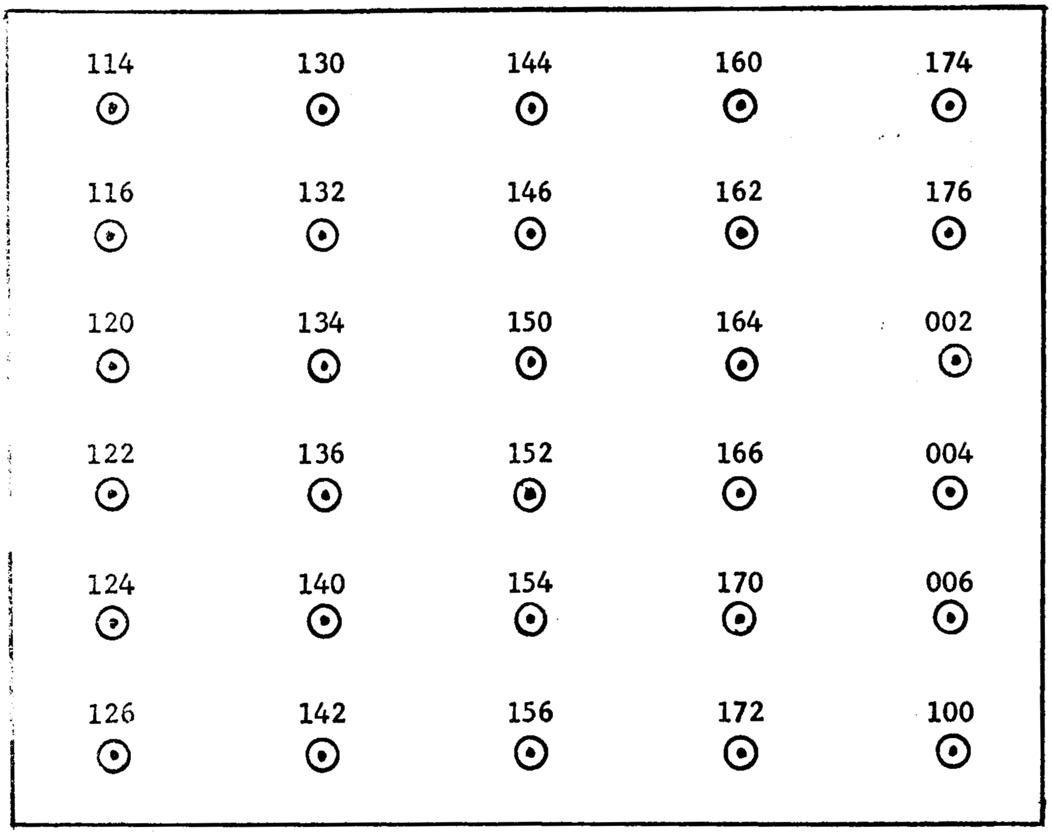
174: X' channel

374: Y' channel

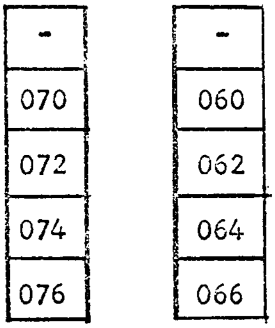
574: Z' channel

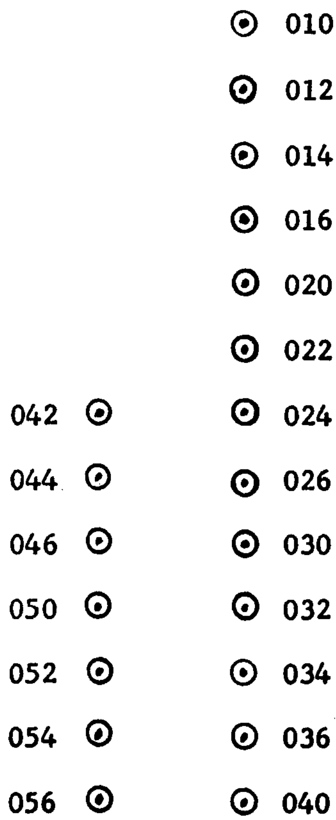
700: I/O channel

C. A "floating" interrupt pushbutton panel is now operational. It is located on the primary display console table. The diagram below shows the interrupt jump sector address associated with each pushbutton on the panel.



The keyboard when in an interrupt mode uses interrupt jump sector address 112. The four function keys to the left of the keyboard utilize sector addresses (from top to bottom) 102, 104, 106 and 110. The following diagrams depict the sector addresses assigned to the interrupt pushbuttons on the display control panel.





OPERATIONS CODES (MAGIC)

OCTAL CODE		
00	BLOCK XFR TO W+X+Y+Z FROM GM OR I/O CHANNELS, SECTOR $\rightarrow$ REV.	
01	BLOCK XFR FROM W+X+Y+Z TO GM OR I/O CHANNELS, SECTOR $\rightarrow$ REV.	
02	" " TO " FROM " " , SPECIFIED SECTOR	
03	" " FROM " TO " " , " "	
10	JUMP: UNCONDITIONAL OR IF SENSE LIGHT (W+X+Y+Z) IS ON	
12	" OVERFLOW IF W+X+Y+Z OFFF ON	
14	" A(W+X+Y+Z) = 0 (SPECIFY SECTOR) (XFR ALT. ADDR. $\rightarrow$ IAC)	
16	" NHFF ON	
11	CONTROL: INTERRUPT	
13	" TRANSFER $LOC_x \rightarrow B_x$, $LOC_y \rightarrow B_y$, KEYBD & ZSW $\rightarrow B_Z$	
15	" EXT. DEVICE (W+X+Y+Z)	
17	" SET SL (W+X+Y+Z), HALT $(W\cdot X\cdot Y\cdot Z)$, BREAKPOINT $(\overline{W}\cdot\overline{X}\cdot\overline{Y}\cdot\overline{Z})$	
21	FILL B(W+X+Y+Z) REGISTER	
22	" C(W+X+Y+Z) REGISTER	
23	" ALT. ADDRESS REGISTER	
24	" MASK REGISTER	
25	" I/O REGISTER	
26	" IAR REGISTER	
31	EMPTY B(W+X+Y+Z) REGISTER	
32	" C(W+X+Y+Z) REGISTER	
33	" ALT. ADDRESS REGISTER	
34	" MASK REGISTER	
35	" I/O REGISTER	
36	" IAR REGISTER	
40	ADD A and C (W+X+Y+Z)	SPECIFIED SECTOR
41	ADD A AND $\overline{C}$ (W+X+Y+Z)	
42	ADD A AND C (W+X+Y+Z) MOD 2	
43	ADD A AND $\overline{C}$ (W+X+Y+Z) MOD 2	
44	" SAME AS ABOVE	SECTOR TO REV.
45	"	
46	"	
47	"	

OPERATIONS CODES (MAGIC)

OCTAL CODE		
50	INSERT CONTENTS OF B(W+X+Y+Z) REGISTER AT SPECIFIED SECTOR	
51	INSERT CONTENTS OF B(W+X+Y+Z) REGISTER AT 1st ZERO DETECTED	
52	DELETE SPECIFIED ADDRESS OF W+X+Y+Z	
53	DELETE AT 1st ZERO DETECTED IN W+X+Y+Z	
60	SCAN (MASKED) $A = C$ (W+X+Y+Z)	SPECIFIED SECTOR TO REV. "HIT" ADDRESS $\longrightarrow$ AAR
61	" $A > C$ "	
62	" $A < C$ "	
63	" $A = 0$ (NOT MASKED) (W+X+Y+Z)	
64	" $A \neq C$ "	
65	" $A = C$, BOTH X AND Y	
66	COMPARE $A = C$ (MASKED, SINGLE SECTOR, W+X+Y+Z)	
70	SHIFT RIGHT SPECIFIED SECTOR	(W+X+Y+Z)
71	" " SECTOR $\longrightarrow$ REV.	
72	SHIFT LEFT SPECIFIED SECTOR	
73	" " SECTOR $\longrightarrow$ REV.	

Alternate addressing not applicable to control instructions (operation codes 11, 13, 15, 17), and the $A = 0$ jump operation (Op. code 14).

MAGIC PROGRAMMING WORKSHEET

PAGE ______

CHANNEL ________

ADDR.	CHANNEL	W	X	OPER.	Y	Z	SECTOR	OPERATION CODE	REMARKS

KEYBOARD CHARACTER CODES FOR MAGIC

CHARACTER	OCTAL CODE	CHARACTER	OCTAL CODE
A	01	No. 0	60
B	02	1	61
C	03	2	62
D	04	3	63
E	05	4	64
F	06	5	65
G	07	6	66
H	10	7	67
I	11	8	70
J	12	9	71
K	13		
L	14	Space	40
M	15	(	50
N	16	)	51
O	17	+ (plus)	53
P	20	- (minus)	55
Q	21	. (period)	56
R	22	* (asterisk)	52
S	23	= (equal)	75
T	24	Upper Case	36
U	25	On 1	57
V	26	Carriage Return	44
W	27	Lower Case	34
X	30		
Y	31		
Z	32		